Œ

intel®

Hof

Pentium® Pro Family Developer's Manual

Volume 1: Specifications

Appendix A Signals Reference

1996

Order Number 242995-001

PATENT NOTICE

Through its investment in computer technology, Intel Corporation (Intel) has acquired numerous proprietary rights, including patents issued by the U.S. Patent and Trademark Office. Intel has patents covering the use or implementation of processors in combination with other products, e.g., certain computer systems. System and method patents or pending patents, of Intel and others, may apply to these systems. A separate license may be required for their use (see Intel Terms and Conditions for details). Specific Intel patents include U.S. patent 4,972,338.

Contact your local Intel sales office or your distributor to obtain the latest specifications before placing your product order.

Copies of documents which have an ordering number and are referenced in this document, or other Intel literature, may be obtained from:

Intel Corporation
P.O. Box 7641
Mt. Prospect, IL 60056-7641

or call 1-800-879-4683

A

Signals Reference

Note: This document is part of the *Pentium® Pro Family Developer's Manual, Volume 1: Specifications*, (Order Number 242690-001) and is intended to be used with that document.

APPENDIX A SIGNALS REFERENCE

This appendix provides an alphabetical listing of all Pentium Pro processor signals. The tables at the end of this appendix summarize the signals by direction: output, input, and I/O.

A.1. ALPHABETICAL SIGNALS REFERENCE

A.1.1. A[35:3]# (I/O)

The A[35:3]# signals are the address signals. They are driven during the two-clock Request Phase by the request initiator. The signals in the two clocks are referenced Aa[35:3]# and Ab[35:3]#. During both clocks, A[35:24]# signals are protected with the AP1# parity signal, and A[23:3]# signals are protected with the AP0# parity signal.

The Aa[35:3]# signals are interpreted based on information carried during the first Request Phase clock on the REQa[4:0]# signals.

For memory transactions as defined by REQa[4:0]# = {XX01X,XX10X,XX11X}, the Aa[35:3]# signals define a 2^{36}-byte physical memory address space. The cacheable agents in the system observe the Aa[35:3]# signals and begin an internal snoop. The memory agents in the system observe the Aa[35:3]# signals and begin address decode to determine if they are responsible for the transaction completion. Aa[4:3]# signals define the critical word, the first data chunk to be transferred on the data bus. Cache line transactions use the burst order described in Section 3.3.4.1., 'Line Transfers" to transfer the remaining three data chunks.

For Pentium Pro processor IO transactions as defined by REQa[4:0]# = 1000X, the signals Aa[16:3]# define a 64K+3 byte physical IO space. The IO agents in the system observe the signals and begin address decode to determine if they are responsible for the transaction completion. Aa[35:17]# are always zero. Aa16# is zero unless the IO space being accessed is the first three bytes of a 64KByte address range.

For deferred reply transactions as defined by REQa[4:0]# = 00000, Aa[23:16]# carry the deferred ID. This signal is the same deferred ID supplied by the request initiator of the original transaction on Ab[23:16]#/DID[7:0]# signals. Pentium Pro processor bus agents that support deferred replies sample the deferred ID and perform an internal match against any outstanding transactions waiting for deferred replies. During a deferred reply, Aa[35:24]# and Aa[15:3]# are reserved.

For the branch-trace message transaction as defined by REQa[4:0]# = 01001 and for special and interrupt acknowledge transactions, as defined by REQa[4:0]# = 01000, the Aa[35:3]# signals are reserved and undefined.

During the second clock of the Request Phase, Ab[35:3]# signals perform identical signal functions for all transactions. For ease of description, these functions are described using new signal names. Ab[31:24]# are renamed the attribute signals ATTR[7:0]#. Ab[23:16]# are renamed the Deferred ID signals DID[7:0]#. Ab[15:8]# are renamed the eight-byte enable signals BE[7:0]#. Ab[7:3]# are renamed the extended function signals EXF[4:0]#.

Ab[31:24]#	Ab[23:16]#	Ab[15:8]#	Ab[7:3]#
ATTR[7:0]#	DID[7:0]#	BE[7:0]#	EXF[4:0]#

On the active-to-inactive transition of RESET#, each Pentium Pro processor bus agent samples A[35:3]# signals to determine its power-on configuration.

A.1.2. A20M# (I)

The A20M# signal is the address-20 mask signal in the PC Compatibility group. If the A20M# input signal is asserted, the Pentium Pro processor masks physical address bit 20 (A20#) before looking up a line in any internal cache and before driving a read/write transaction on the bus. Asserting A20M# emulates the 8086 processor's address wrap-around at the one Mbyte boundary. Only assert A20M# when the processor is in real mode. The effect of asserting A20M# in protected mode is undefined and may be implemented differently in future processors.

Snoop requests and cache-line writeback transactions are unaffected by A20M# input. Address 20 is not masked when the processor samples external addresses to perform internal snooping.

A20M# is an asynchronous input. However, to guarantee recognition of this signal following an I/O write instruction, A20M# must be valid with active RS[2:0]# signals of the corresponding I/O Write bus transaction. In FRC mode, A20M# must be synchronous to BCLK.

During active RESET#, the Pentium Pro processor begins sampling the A20M#, IGNNE# , and LINT[1:0] values to determine the ratio of core-clock frequency to bus-clock frequency. See Table 9-4. After the PLL-lock time, the core clock becomes stable and is locked to the external bus clock. On the active-to-inactive transition of RESET#, the Pentium Pro processor latches A20M#, IGNNE#, and LINT[1:0] and freezes the frequency ratio internally.

A.1.3. ADS# (I/O)

The ADS# signal is the address Strobe signal. It is asserted by the current bus owner for one clock to indicate a new Request Phase. A new Request Phase can only begin if the In-order Queue has less than the maximum number of entries defined by the power-on configuration (1 or 8), the Request Phase is not being stalled by an active BNR# sequence and the ADS# associated with the previous Request Phase is sampled inactive. Along with the ADS#, the request initiator drives A[35:3]#, REQ[4:0]#, AP[1:0]#, and RP# signals for two clocks. During the second Request Phase clock, ADS# must be inactive. RP# provides parity protection for REQ[4:0]# and ADS# signals during both clocks. If the transaction is part of a bus locked operation, LOCK# must be active with ADS#.

If the request initiator continues to own the bus after the first Request Phase, it can issue a new request every three clocks. If the request initiator needs to release the bus ownership after the Request Phase, it can deactivate its BREQ*n*#/ BPRI# arbitration signal as early as with the activation of ADS#.

All bus agents observe the ADS# activation to begin parity checking, protocol checking, address decode, internal snoop, or deferred reply ID match operations associated with the new transaction. On sampling the asserted ADS#, all agents load the new transaction in the In-order Queue and update internal counters. The Error, Snoop, Response, and Data Phase of the transaction are defined with respect to ADS# assertion.

A.1.4. AERR# (I/O)

The AERR# signal is the address parity error signal. Assuming the AERR# driver is enabled during the power-on configuration, a bus agent can drive AERR# active for exactly one clock during the Error Phase of a transaction. AERR# must be inactive for a minimum of two clocks. The Error Phase is always three clocks from the beginning of the Request Phase.

On observing active ADS#, all agents begin parity and protocol checks for the signals valid in the two Request Phase clocks. Parity is checked on AP[1:0]# and RP# signals. AP1# protects A[35:24]#, AP0# protects A[23:3]# and RP# protects REQ[4:0]#. A parity error without a protocol violation is signalled by AERR# assertion.

If AERR# observation is enabled during power-on configuration, AERR# assertion in a valid Error Phase aborts the transaction. All bus agents remove the transaction from the In-order Queue and update internal counters. The Snoop Phase, Response Phase, and Data Phase of the transaction are aborted. All signals in these phases must be deasserted two clocks after AERR# is asserted, even if the signals have been asserted before AERR# has been observed. Specifically if the Snoop Phase associated with the aborted transaction is driven in the next clock, the snoop results, including a STALL condition (HIT# and HITM# asserted for one clock), are ignored. All bus agents must also begin an arbitration reset sequence and deassert BREQ*n*#/BPRI# arbitration signals on sampling AERR# active. A current bus owner in the middle of a bus lock operation must keep LOCK# asserted and assert its arbitration request BPRI#/BREQ*n*# after keeping it inactive for two clocks to retain its bus ownership and guarantee lock atomicity. All other agents, including the current bus owner not in the middle of a bus lock operation, must wait at least 4 clocks before asserting BPRI#/BREQ*n*# and beginning a new arbitration.

If AERR# observation is enabled, the Pentium Pro processor retries the transaction once. After a single retry, the request initiator treats the error as a hard error and asserts BERR# or enters the Machine Check Exception handler, as defined by the system configuration.

If AERR# observation is disabled during power-on configuration, AERR# assertion is ignored by all bus agents except a central agent. Based on the Machine Check Architecture of the system, the central agent can ignore AERR#, assert NMI to execute NMI handler, or assert BINIT# to reset the bus units of all agents and execute an MCE handler.

A.1.5. AP[1:0]# (I/O)

The AP[1:0]# signals are the address parity signals. They are driven by the request initiator during the two Request Phase clocks along with ADS#, A[35:3]#, REQ[4:0]#, and RP#. AP1# covers A[35:24]#. AP0# covers A[23:3]#. A correct parity signal is high if an even number of covered signals are low and low if an odd number of covered signals are low. This rule allows parity to be high when all the covered signals are high.

Provided "AERR# drive" is enabled during the power-on configuration, all bus agents begin parity checking on observing active ADS# and determine if there is a parity error. On observing a parity error on any one of the two Request Phase clocks, the bus agent asserts AERR# during the Error Phase of the transaction.

A.1.6. ASZ[1:0]# (I/O)

The ASZ[1:0]# signals are the memory address-space size signals. They are driven by the request initiator during the first Request Phase clock on the REQa[4:3]# pins. The ASZ[1:0]# signals are valid only when REQa[1:0]# signals equal 01B, 10B, or 11B, indicating a memory access transaction. The ASZ[1:0]# decode is defined in Table A-1.

Table A-1. ASZ[1:0]# Signal Decode

ASZ[1:0]#		Description
0	0	0 <= A[35:3]# < 4 GB
0	1	4 GB <= A[35:3]# < 64 GB
1	x	Reserved

If the memory access is within the 0-to-(4GByte -1) address space, ASZ[1:0]# must be 00B. If the memory access is within the 4Gbyte-to-(64GByte -1) address space, ASZ[1:0]# must be 01B. All observing bus agents that support the 4Gbyte (32 bit) address space must respond to the transaction only when ASZ[1:0]# equals 00. All observing bus agents that support the 64GByte (36- bit) address space must respond to the transaction when ASZ[1:0]# equals 00B or 01B.

A.1.7. ATTR[7:0]# (I/O)

The ATTR[7:0]# signals are the attribute signals. They are driven by the request initiator during the second Request Phase clock on the Ab[31:24]# pins. The ATTR[7:0]# signals are valid for all transactions. The ATTR[7:3]# are reserved and undefined. The ATTR[2:0]# are driven based on the Memory Range Register attributes and the Page Table attributes. Table A-2 defines ATTR[3:0]# signals.

Table A-2. ATTR[7:0]# Field Descriptions

ATTR[7:3]#	ATTR[2]#	ATTR[1:0]#			
Reserved (0)	Potentially Speculatable	11	10	01	00
		WriteBack	WriteProtect	WriteThrough	UnCacheable

A.1.8. BCLK (I)

The BCLK (clock) signal is the Execution Control group input signal. It determines the bus frequency. All agents drive their outputs and latch their inputs on the BCLK rising edge.

The BCLK signal indirectly determines the Pentium Pro processor's internal clock frequency. Each Pentium Pro processor derives its internal clock from BCLK by multiplying the BCLK frequency by 2, 3, or 4 as defined and allowed by the power-on configuration.

All external timing parameters are specified with respect to the BCLK signal.

A.1.9. BE[7:0]# (I/O)

The BE[7:0]# signals are the byte-enable signals. They are driven by the request initiator during the second Request Phase clock on the Ab[15:8]# pins. These signals carry various information depending on the REQ[4:0]# value.

For memory or I/O transactions (REQa[4:0]# = {10000B, 10001B, XX01XB, XX10XB, XX11XB}) the byte-enable signals indicate that valid data is requested or being transferred on the corresponding byte on the 64 bit data bus. BE0# indicates D[7:0]# is valid, BE1# indicates D[15:8]# is valid, ..., BE7# indicates D[63:56]# is valid.

For Special transactions ((REQa[4:0]# = 01000B) and (REQb[1:0]# = 01B)), the BE[7:0]# signals carry special cycle encodings as defined in Table A-3. All other encodings are reserved.

Table A-3. Special Transaction Encoding on BE[7:0]#

BE[7:0]#	Special Cycle
0000 0000	Reserved
0000 0001	Shutdown
0000 0010	Flush
0000 0011	Halt
0000 0100	Sync
0000 0101	Flush Acknowledge
0000 0110	Stop Clock Acknowledge
0000 0111	SMI Acknowledge
0000 1000 through 1111 1111	Reserved

For Deferred Reply, Interrupt Acknowledge, and Branch Trace Message transactions, the BE[7:0]# signals are undefined.

A.1.10. BERR# (I/O)

The BERR# signal is the Error group Bus Error signal. It is asserted to indicate an unrecoverable error without a bus protocol violation.

The BERR# protocol is as follows: If an agent detects an unrecoverable error for which BERR# is a valid error response and BERR# is sampled inactive, it asserts BERR# for three clocks. An agent can assert BERR# only after observing that the signal is inactive. An agent asserting BERR# must deassert the signal in two clocks if it observes that another agent began asserting BERR# in the previous clock.

BERR# assertion conditions are defined by the system configuration. Configuration options enable the BERR# driver as follows:

- enabled or disabled
- asserted optionally for internal errors along with IERR#
- optionally asserted by the request initiator of a bus transaction after it observes an error
- asserted by any bus agent when it observes an error in a bus transaction

BERR# *sampling* conditions are also defined by the system configuration. Configuration options enable the BERR# receiver to be enabled or disabled. When the bus agent samples an active BERR# signal and if MCE is enabled, the Pentium Pro processor enters the Machine Check Handler. If MCE is disabled, typically the central agent forwards BERR# as an NMI to one of the processors. The Pentium Pro processor does not support BERR# sampling (always disabled).

A.1.11. BINIT# (I/O)

The BINIT# signal is the bus initialization signal. If the BINIT# driver is enabled during the power on configuration, BINIT# is asserted to signal any bus condition that prevents reliable future information.

The BINIT# protocol is as follows: If an agent detects an error for which BINIT# is a valid error response, and BINIT# is sampled inactive, it asserts BINIT# for three clocks. An agent can assert BINIT# only after observing that the signal is inactive. An agent asserting BINIT# must deassert the signal in two clocks if it observes that another agent began asserting BINIT# in the previous clock.

If BINIT# observation is enabled during power-on configuration, and BINIT# is sampled asserted, all bus state machines are reset. All agents reset their rotating ID for bus arbitration to the state after reset, and internal count information is lost. The L1 and L2 caches are not affected.

If BINIT# observation is disabled during power-on configuration, BINIT# is ignored by all bus agents except a central agent that must handle the error in a manner appropriate to the system architecture.

A.1.12. BNR# (I/O)

The BNR# signal is the Block Next Request signal in the Arbitration group. The BNR# signal is used to assert a bus stall by any bus agent who is unable to accept new bus transactions to avoid an internal transaction queue overflow. During a bus stall, the current bus owner cannot issue any new transactions.

Since multiple agents might need to request a bus stall at the same time, BNR# is a wire-OR signal. In order to avoid wire-OR glitches associated with simultaneous edge transitions driven by multiple drivers, BNR# is activated on specific clock edges and sampled on specific clock edges. A valid bus stall involves assertion of BNR# for one clock on a well-defined clock edge (T1), followed by de-assertion of BNR# for one clock on the next clock edge (T1+1). BNR# can first be sampled on the second clock edge (T1+1) and must always be ignored on the third clock edge (T1+2). An extension of a bus stall requires one clock active (T1+2), one clock inactive (T1+3) BNR# sequence with BNR# sampling points every two clocks (T1+1, T1+3,...).

After the RESET# active-to-inactive transition, bus agents might need to perform hardware initialization of their bus unit logic. Bus agents intending to create a request stall must assert BNR# in the clock after RESET# is sampled inactive.

After BINIT# assertion, all bus agents go through a similar hardware initialization and can create a request stall by asserting BNR# four clocks after BINIT# assertion is sampled.

On the first BNR# sampling clock that BNR# is sampled inactive, the current bus owner is allowed to issue one new request. Any bus agent can immediately reassert BNR# (four clocks from the previous assertion or two clocks from the previous de-assertion) to create a new bus stall. This throttling mechanism enables independent control on every new request generation.

If BNR# is deasserted on two consecutive sampling points, new requests can be freely generated on the bus. After receiving a new transaction, a bus agent can require an address stall due to an anticipated transaction-queue overflow condition. In response, the bus agent can assert BNR#, three clocks from active ADS# assertion and create a bus stall. Once a bus stall is created, the bus remains stalled until BNR# is sampled asserted on subsequent sampling points.

A.1.13. BP[3:2]# (I/O)

The BP[3:2]# signals are the System Support group Breakpoint signals. They are outputs from the Pentium Pro processor that indicate the status of breakpoints.

A.1.14. BPM[1:0]# (I/O)

The BPM[1:0]# signals are more System Support group breakpoint and performance monitor signals. They are outputs from the Pentium Pro processor that indicate the status of breakpoints and programmable counters used for monitoring Pentium Pro processor performance.

A.1.15. BPRI# (I)

The BPRI# signal is the Priority-agent Bus Request signal. The priority agent arbitrates for the bus by asserting BPRI#. The priority agent is always be the next bus owner. Observing BPRI# active causes the current symmetric owner to stop issuing new requests, unless such requests are part of an ongoing locked operation.

If LOCK# is sampled inactive two clocks from BPRI# driven asserted, the priority agent can issue a new request within four clocks of asserting BPRI#. The priority agent can further reduce its arbitration latency to two clocks if it samples active ADS# and inactive LOCK# on the clock in which BPRI# was driven active and to three clocks if it samples active ADS# and inactive LOCK# on the clock in which BPRI# was sampled active. If LOCK# is sampled active, the priority agent must wait for LOCK# deasserted and gains bus ownership in two clocks after LOCK# is sampled deasserted. The priority agent can keep BPRI# asserted until all of its requests are completed and can release the bus by de-asserting BPRI# as early as the same clock edge on which it issues the last request.

On observation of active AERR#, RESET#, or BINIT#, BPRI# must be deasserted in the next clock. BPRI# can be reasserted in the clock after sampling the RESET# active-to-inactive transition or three clocks after sampling BINIT# active and RESET# inactive. On AERR# assertion, if the priority agent is in the middle of a bus-locked operation, BPRI# must be re-asserted after two clocks, otherwise BPRI# must stay inactive for at least 4 clocks.

After the RESET# inactive transition, Pentium Pro processor bus agents begin BPRI# and BNR# sampling on BNR# sample points. When both BNR# and BPRI# are observed inactive on a BNR# sampling point, the APIC units in Pentium Pro processors on a common APIC bus are synchronized. In a system with multiple Pentium Pro processor bus clusters sharing a common APIC bus, BPRI# signals of all clusters must be asserted after RESET# until BNR# is observed inactive on a BNR# sampling point. The BPRI# signal on all Pentium Pro processor buses must then be deasserted within 100ns of each other to accomplish APIC bus synchronization across all processors.

A.1.16. BR0#(I/O), BR[3:1]# (I)

The BR[3:0]# pins are the physical bus request pins that drive the BREQ[3:0]# signals in the system. The BREQ[3:0]# signals are interconnected in a rotating manner to individual processor pins. #. Table A-4 gives the rotating interconnect between the processor and bus signals.

Table A-4. BR0#(I/O), BR1#, BR2#, BR3# Signals Rotating Interconnect

Bus Signal	Agent 0 Pins	Agent 1 Pins	Agent 2 Pins	Agent 3 Pins
BREQ0#	BR0#	BR3#	BR2#	BR1#
BREQ1#	BR1#	BR0#	BR3#	BR2#
BREQ2#	BR2#	BR1#	BR0#	BR3#
BREQ3#	BR3#	BR2#	BR1#	BR0#

During power-up configuration, the central agent must assert the BR0# bus signal. All symmetric agents sample their BR[3:0]# pins on active-to-inactive transition of RESET#. The pin on which the agent samples an active level determines its agent ID. All agents then configure their pins to match the appropriate bus signal protocol, as shown in Table A-5.

Table A-5. BR[3:0]# Signal Agent IDs

Pin Sampled Active on RESET#	Agent ID
BR0#	0
BR3#	1
BR2#	2
BR1#	3

A.1.17. BREQ[3:0]# (I/O)

The BREQ[3:0]# signals are the Symmetric-agent Arbitration Bus signals (called bus request). A symmetric agent *n* arbitrates for the bus by asserting its BREQ*n*# signal. Agent *n* drives BREQ*n*# as an output and receives the remaining BREQ[3:0]# signals as inputs.

The symmetric agents support distributed arbitration based on a round-robin mechanism. The rotating ID is an internal state used by all symmetric agents to track the agent with the lowest priority at the next arbitration event. At power-on, the rotating ID is initialized to three, allowing agent 0 to be the highest priority symmetric agent. After a new arbitration event, the rotating ID of all symmetric agents is updated to the agent ID of the symmetric owner. This update gives the new symmetric owner lowest priority in the next arbitration event.

A new arbitration event occurs either when a symmetric agent asserts its BREQ*n*# on an Idle bus (all BREQ[3:0]# previously inactive), or the current symmetric owner de-asserts BREQ*m*# to release the bus ownership to a new bus owner *n*. On a new arbitration event, based on BREQ[3:0]#, and the rotating ID, all symmetric agents simultaneously determine the new symmetric owner. The symmetric owner can park on the bus (hold the bus) provided that no other symmetric agent is requesting its use. The symmetric owner parks by keeping its BREQ*n*# signal active. On sampling active BREQ*m*# asserted by another symmetric agent, the symmetric owner de-asserts BREQ*n*# as soon as possible to release the bus. A symmetric owner stops issuing new requests that are not part of an existing locked operation upon observing BPRI# active.

A symmetric agent can not deassert BREQ*n*# until it becomes a symmetric owner. A symmetric agent can reassert BREQ*n*# after keeping it inactive for one clock.

On observation of active AERR#, RESET#, or BINIT#, the BREQ[3:0]# signals must be deasserted in the next clock. BREQ[3:0]# can be reasserted in the clock after sampling the RESET# active-to-inactive transition or three clocks after sampling BINIT# active and RESET# inactive. On AERR# assertion, if bus agent n is in the middle of a bus-locked operation, BREQn# must be re-asserted after two clocks, otherwise BREQ[3:0]# must stay inactive for at least 4 clocks.

A.1.18. D[63:0]# (I/O)

The D[63:0]# signals are the data signals. They are driven during the Data Phase by the agent responsible for driving the data. These signals provide a 64-bit data path between various Pentium Pro processor bus agents. 32-byte line transfers require four data transfer clocks with valid data on all eight bytes. Partial transfers require one data transfer clock with valid data on the byte(s) indicated by active byte enables BE[7:0]#. Data signals not valid for a particular transfer must still have correct ECC (if data bus ECC is selected). If BE0# is asserted, D[7:0]# transfers the least significant byte. If BE7# is asserted, D[63:56]# transfers the most significant byte.

The data driver asserts DRDY# to indicate a valid data transfer. If the Data Phase involves more than one clock the data driver also asserts DBSY# at the beginning of the Data Phase and de-asserts DBSY# no earlier than on the same clock that it performs the last data transfer.

A.1.19. DBSY# (I/O)

The DBSY# signal is the Data-bus Busy signal. It indicates that the data bus is busy. It is asserted by the agent responsible for driving the data during the Data Phase, provided the Data Phase involves more than one clock. DBSY# is asserted at the beginning of the Data Phase and may be deasserted on or after the clock on which the last data is driven. The data bus is released one clock after DBSY# is deasserted.

When normal read data is being returned, the Data Phase begins with the Response Phase. Thus the agent returning read data can assert DBSY# when the transaction reaches the top of the In-order Queue and it is ready to return response on RS[2:0]# signals. In response to a write request, the agent driving the write data must drive DBSY# active after the write transaction reaches the top of the In-order Queue and it sees active TRDY# with inactive DBSY# indicating that the target is ready to receive data. For an implicit writeback response, the snoop agent must assert DBSY# active after the target memory agent of the implicit writeback asserts TRDY#. Implicit writeback TRDY# assertion begins after the transaction reaches the top of the In-order Queue, and TRDY# de-assertion associated with the write portion of the transaction, if any is completed. In this case, the memory agent guarantees assertion of implicit writeback response in the same clock in which the snooping agent asserts DBSY#.

A.1.20. DEFER# (I)

The DEFER# signal is the defer signal. It is asserted by an agent during the Snoop Phase to indicate that the transaction cannot be guaranteed in-order completion. Assertion of DEFER# is normally the responsibility of the addressed memory agent or I/O agent. For systems that involve resources on a system bus other than the Pentium Pro processor bus, a bridge agent can accept the DEFER# assertion responsibility on behalf of the addressed agent.

When HITM# and DEFER# are both active during the Snoop Phase, HITM# is given priority and the transaction must be completed with implicit writeback response. If HITM# is inactive, and DEFER# active, the agent asserting DEFER# must complete the transaction with a Deferred or Retry response.

If DEFER# is inactive, or HITM# is active, then the transaction is committed for in-order completion and snoop ownership is transferred normally between the requesting agent, the snooping agents, and the response agent.

If DEFER# is active with HITM# inactive, the transaction commitment is deferred. If the defer agent completes the transaction with a retry response, the requesting agent must retry the transaction. If the defer agent returns a deferred response, the requesting agent must freeze snoop state transitions associated with the deferred transaction and issues of new order-dependent transactions until the corresponding deferred reply transaction. In the meantime, the ownership of the deferred address is transferred to the defer agent and it must guarantee management of conflicting transactions issued to the same address.

If DEFER# is active in response to a newly issued bus-lock transaction, the entire bus-locked operation is re-initiated regardless of HITM#. This feature is useful for a bridge agent in response to a split bus-locked operation. It is recommended that the bridge agent extend the Snoop Phase of the first transaction in a split locked operation until it can either guarantee ownership of all system resources to enable successful completion of the split sequence or assert DEFER# followed by a Retry Response to abort the split sequence.

A.1.21. DEN# (I/O)

The DEN# signal is the defer-enable signal. It is driven to the bus on the second clock of the Request Phase on the EXF1#/Ab4# pin. DEN# is asserted to indicate that the transaction can be deferred by the responding agent.

A.1.22. DEP[7:0]# (I/O)

The DEP[7:0]# signals are the data bus ECC protection signals. They are driven during the Data Phase by the agent responsible for driving D[63:0]#. The DEP[7:0]# signals provide optional ECC protection for the data bus. During power-on configuration, DEP[7:0]# signals can be enabled for either ECC checking or no checking.

The ECC error correcting code can detect and correct single-bit errors and detect double-bit or nibble errors. Chapter 8, *Data Integrity* provides more information about ECC.

DEP[7:0]# provide valid ECC for the entire data bus on each data clock, regardless of which bytes are valid. If checking is enabled, receiving agents check the ECC signals for all 64 data signals.

A.1.23. DID[7:0]# (I/O)

The DID[7:0]# signals are Deferred Identifier signals. They are transferred using A[23:16]# signals by the request initiator. They are transferred on Ab[23:16]# during the second clock of the Request Phase on all transactions, but only defined for deferrable transactions (DEN# asserted). DID[7:0]# is also transferred on Aa[23:16]# during the first clock of the Request Phase for Deferred Reply tranactions.

The deferred identifier defines the token supplied by the request initiator. DID[7:4]# carry the request initiators' agent identifier and DID[3:0]# carry a transaction identifier associated with the request. This configuration limits the bus specification to 16 bus masters with each one of the bus masters capable of making up to sixteen requests.

Every deferrable transaction issued on the Pentium Pro processor bus which has not been guaranteed completion (has not successfully passed its Snoop Result Phase) will have a unique Deferred ID. This includes all outstanding transactions which have not had their snoop result reported, or have had their snoop results deferred. After a deferrable transaction passes its Snoop Result Phase without DEFER# asserted, its Deferred ID may be reused. Similarly, the deferred ID of a transaction which was deferred may be reused after the completion of the snoop window of the deferred reply.

DID[7]# indicates the agent type. Symmetric agents use 0. Priority agents use 1. DID[6:4]# indicates the agent ID. Symmetric agents use their arbitration ID. The Pentium Pro processor has four symmetric agents, so does not assert DID[6]#. DID[3:0]# indicates the transaction ID for an agent. The transaction ID must be unique for all transactions issued by an agent which have not reported their snoop results.

Table A-6. DID[7:0]# Encoding

DID[7]#	DID[6:4]#	DID[3:0]#
Agent Type	Agent ID	Transaction ID

The Deferred Reply agent transmits the DID[7:0]# (Ab[23:16]#) signals received during the original transaction on the Aa[23:16]# signals during the Deferred Reply transaction. This process enables the original request initiator to make an identifier match and wake up the original request waiting for completion.

A.1.24. DRDY# (I/O)

The DRDY# signal is the Data Phase data-ready signal. The data driver asserts DRDY# on each data transfer, indicating valid data on the data bus. In a multi-cycle data transfer, DRDY# can be deasserted to insert idle clocks in the Data Phase. During a line transfer, DRDY# is active for four clocks. During a partial 1-to-8 byte transfer, DRDY# is active for one clock. If a data transfer is exactly one clock, then the entire Data Phase may consist of only one clock active DRDY# and inactive DBSY#. If DBSY# is asserted for a 1-to-8 byte transfer, then the data bus is not released until one clock after DBSY# is deasserted.

A.1.25. DSZ[1:0]# (I/O)

The DSZ[1:0]# signals are the data-size signals. They are transferred on REQb[4:3]# signals in the second clock of Request Phase by the requesting agent. The DSZ[1:0]# signals define the data transfer capability of the requesting agent. For the Pentium Pro processor, DSZ#= 00, always.

A.1.26. EXF[4:0]# (I/O)

The EXF[4:0]# signals are the Extended Function signals. They are transferred on the Ab[7:3]# signals by the request initiator during the second clock of the Request Phase. The signals specify any special functional requirement associated with the transaction based on the requestor mode or capability. The signals are defined in Table A-7.

Table A-7. EFX[4:0]# Signal Definitions

EXF	Name	Extended Functionality	When Activated
EXF4#	SMMEM#	SMM Mode	After entering SMM mode
EXF3#	SPLCK#	Split Lock	The first transaction of a split bus lock operation
EXF2#	Reserved	Reserved	
EXF1#	DEN#	Defer Enable	The transactions for which Defer or Retry Response is acceptable.
EXF0#	Reserved	Reserved	

A.1.27. FERR# (O)

The FERR# signal is the PC Compatibility group Floating-point Error signal. The Pentium Pro processor asserts FERR# when it detects an unmasked floating-point error. FERR# is similar to the ERROR# signal on the Intel387™ coprocessor. FERR# is included for compatibility with systems using DOS-type floating-point error reporting.

A.1.28. FLUSH# (I)

When the FLUSH# input signal is asserted, the Pentium Pro processor bus agent writes back all internal cache lines in the Modified state and invalidates all internal cache lines. At the completion of a flush operation, the Pentium Pro processor issues a Flush Acknowledge transaction to indicate that the cache flush operation is complete. The Pentium Pro processor stops caching any new data while the FLUSH# signal remains asserted.

FLUSH# is an asynchronous input. However, to guarantee recognition of this signal following an I/O write instruction, FLUSH# must be valid along with RS[2:0]# in the Response Phase of the corresponding I/O Write bus transaction. In FRC mode, FLUSH# must be synchronous to BCLK.

On the active-to-inactive transition of RESET#, each Pentium Pro processor bus agent samples FLUSH# to determine its power-on configuration. See Table 9-4.

A.1.29. FRCERR(I/O)

The FRCERR signal is the Error group Functional-redundancy-check Error signal. If two Pentium Pro processors are configured in an FRC pair, as a single "logical" processor, then the checker processor asserts FRCERR if it detects a mismatch between its internally sampled outputs and the master processor's outputs. The checker's FRCERR output pin is connected to the master's FRCERR input pin.

For point-to-point connections, the checker always compares against the master's outputs. For bussed single-driver signals, the checker compares against the signal when the master is the only allowed driver. For bussed multiple-driver Wire-OR signals, the checker compares against the signal only if the master is expected to drive the signal low.

FRCERR is also toggled during the Pentium Pro processor's reset action. A Pentium Pro processor asserts FRCERR for approximately 1 second after RESET's active-to-inactive transition if it executes its built-in self-test (BIST). When BIST execution completes, the Pentium Pro processor de-asserts FRCERR if BIST completed successfully and continues to assert FRCERR if BIST fails. If the Pentium Pro processor does not execute the BIST action, then it keeps FRCERR asserted for approximately 20 clocks and then de-asserts it.

Chapter 9, *Configuration* describes how a Pentium Pro processor can be configured as a master or a checker.

A.1.30. HIT# (I/O), HITM#(I/O)

The HIT# and HITM# signals are Snoop-hit and Hit-modified signals. They are snoop results asserted by any Pentium Pro processor bus agent in the Snoop Phase.

Any bus agent can assert both HIT# and HITM# together for one clock in the Snoop Phase to indicate that it requires a snoop stall. When a stall condition is sampled, all bus agents extend the Snoop Phase by two clocks. The stall can be continued by reasserting HIT# and HITM# together every other clock for one clock.

A caching agent must assert HITM# for one clock in the Snoop Phase if the transaction hits a Modified line, and the snooping agent must perform an implicit writeback to update main memory. The snooping agent with the Modified line makes a transition to Shared state if the original transaction is Read Line or Read Partial, otherwise it transitions to Invalid state. A Deferred Reply transaction may have HITM# asserted to indicate the return of unexpected data.

A snooping agent must assert HIT# for one clock during the Snoop Phase if the line does not hit a Modified line in its writeback cache and if at the end of the transaction it plans to keep the line in Shared state. Multiple caching agents can assert HIT# in the same Snoop Phase. If the requesting agent observes HIT# active during the Snoop Phase it can not cache the line in Exclusive or Modified state.

On observing a snoop stall, the agents asserting HIT# and HITM# independently reassert the signal after one inactive clock so that the correct snoop result is available, in case the Snoop Phase terminates after the two clock extension.

A.1.31. IERR# (O)

The IERR# signal is the Error group Internal Error signal. A Pentium Pro processor asserts IERR# when it observes an internal error. It keeps IERR# asserted until it is turned off as part of the Machine Check Error or the NMI handler in software, or with RESET#, BINIT#, and INIT# assertion.

An internal error can be handled in several ways inside the processor based on its power-on configuration. If Machine Check Exception (MCE) is enabled, IERR# causes an MCE entry. IERR# can also be directed on the BERR# pin to indicate an error. Usually BERR# is sampled back by all processors to enter MCE or it can be redirected as an NMI by the central agent.

A.1.32. IGNNE# (I)

The IGNNE# signal is the PC Compatibility group Ignore Numeric Error signal. If IGNNE# is asserted, the Pentium Pro processor ignores a numeric error and continues to execute non-control floating-point instructions. If IGNNE# is deasserted, the Pentium Pro processor freezes on a non-control floating-point instruction if a previous instruction caused an error.

IGNNE# has no effect when the NE bit in control register 0 is set.

IGNNE# is an asynchronous input. However, to guarantee recognition of this signal following an I/O write instruction, IGNNE# must be valid along with RS[2:0]# in the Response Phase of the corresponding I/O Write bus transaction. In FRC mode, IGNNE# must be synchronous to BCLK.

During active RESET#, the Pentium Pro processor begins sampling the A20M# , IGNNE# and LINT[1:0] values to determine the ratio of core-clock frequency to bus-clock frequency. See Table 9-4. After the PLL-lock time, the core clock becomes stable and is locked to the external bus clock. On the active-to-inactive transition of RESET#, the Pentium Pro processor latches the ratio and freezes the frequency ratio internally.

A.1.33. INIT# (I)

The INIT# signal is the Execution Control group initialization signal. Active INIT# input resets integer registers inside all Pentium Pro processors without affecting their internal (L1 or L2) caches or their floating-point registers. Each Pentium Pro processor begins execution at the power-on reset vector configured during power-on configuration regardless of whether INIT# has gone inactive. The processor continues to handle snoop requests during INIT# assertion.

INIT# can be used to help performance of DOS extenders written for the Intel 80286 processor. INIT# provides a method to switch from protected mode to real mode while maintaining the contents of the internal caches and floating-point state. INIT# can not be used in lieu of RESET# after power-up.

On active-to-inactive transition of RESET#, each Pentium Pro processor bus agent samples INIT# signals to determine its power-on configuration.

INIT# is an asynchronous input. In FRC mode, INIT# must be synchronous to BCLK.

A.1.34. INTR (I)

The INTR signal is the Interrupt Request signal. The INTR input indicates that an external interrupt has been generated. The interrupt is maskable using the IF bit in the EFLAGS register. If the IF bit is set, the Pentium Pro processor vectors to the interrupt handler after the current instruction execution is completed. Upon recognizing the interrupt request, the Pentium Pro processor issues a single Interrupt Acknowledge (INTA) bus transaction. INTR must remain active until the INTA bus transaction to guarantee its recognition.

INTR is sampled on every rising BCLK edge. INTR is an asynchronous input but recognition of INTR is guaranteed in a specific clock if it is asserted synchronously and meets the setup and hold times. INTR must also be deasserted for a minimum of two clocks to guarantee its inactive recognition. In FRC mode, INTR must be synchronous to BCLK. On power-up the LINT[1:0] signals are used for power-on-configuration of clock ratios. Both these signals must be software configured by programming the APIC register space to be used either as NMI/INTR or LINT[1:0] in the BIOS. Because APIC is enabled after reset, LINT[1:0] is the default configuration.

A.1.35. LEN[1:0]# (I/O)

The LEN[1:0]# signals are data-length signals. They are transmitted using REQb[1:0]# signals by the request initiator in the second clock of Request Phase. LEN[1:0]# define the length of the data transfer requested by the request initiator as defined in Table A-8. The LEN[1:0]#, HITM#, and RS[2:0]# signals together define the length of the actual data transfer.

Table A-8. LEN[1:0]# Signals Data Transfer Lengths

LEN[1:0]#	Request Initiator's Data Transfer Length
00	0-8 Bytes
01	16 Bytes
10	32 Bytes
11	Reserved

A.1.36. LINT[1:0] (I)

The LINT[1:0] signals are the Execution Control group Local Interrupt signals. When APIC is disabled, the LINT0 signal becomes INTR, a maskable interrupt request signal, and LINT1 becomes NMI, a non-maskable interrupt. INTR and NMI are backward compatible with the same signals for the Pentium processor. Both signals are asynchronous inputs. In FRC mode, LINT[1:0] must be synchronous to BCLK.

During active RESET#, the Pentium Pro processor continuously samples the A20M#, IGNNE#, and LINT[1:0] values to determine the ratio of core-clock frequency to bus-clock frequency. After the PLL-lock time, the core clock becomes stable and is locked to the external bus clock. On the active-to-inactive transition of RESET#, the Pentium Pro processor freezes the frequency ratio internally.

Both of these signals must be software configured by programming the APIC register space to be used either as NMI/INTR or LINT[1:0] in the BIOS. Because APIC is enabled after reset, LINT[1:0] is the default configuration.

A.1.37. LOCK# (I/O)

The LOCK# signal is the Arbitration group bus lock signal. For a locked sequence of transactions, LOCK# is asserted from the first transaction's Request Phase through the last transaction's Response Phase. A locked operation can be prematurely aborted (and LOCK# deasserted) if AERR# or DEFER# is asserted during the first bus transaction of the sequence. The sequence can also be prematurely aborted if a hard error (such as a hard failure response or AERR# assertion beyond the retry limit) occurs on any one of the transactions during the locked operation.

When the priority agent asserts BPRI# to arbitrate for bus ownership, it waits until it observes LOCK# deasserted. This enables symmetric agents to retain bus ownership throughout the bus locked operation and guarantee the atomicity of lock. If AERR# is asserted up to the retry limit during an ongoing locked operation, the arbitration protocol ensures that the lock owner receives the bus ownership after arbitration logic is reset. This result is accomplished by requiring the lock owner to reactivate its arbitration request one clock ahead of other agents' arbitration request. LOCK# is kept asserted throughout the arbitration reset sequence.

A.1.38. NMI (I)

The NMI signal is the Non-maskable Interrupt signal. It is the state of the LINT1 signal when APIC is disabled. Asserting NMI causes an interrupt with an internally supplied vector value of 2. An external interrupt-acknowledge transaction is not generated. If NMI is asserted during the execution of an NMI service routine, it remains pending and is recognized after the IRET is executed by the NMI service routine. At most, one assertion of NMI is held pending.

NMI is rising-edge sensitive. Recognition of NMI is guaranteed in a specific clock if it is asserted synchronously and meets the setup and hold times. If asserted asynchronously, active and inactive pulse widths must be a minimum of two clocks. In FRC mode, NMI must be synchronous to BCLK.

A.1.39. PICCLK (I)

The PICCLK signal is the Execution Control group APIC Clock signal. It is an input clock to the Pentium Pro processor for synchronous operation of the APIC bus. PICCLK must be synchronous to BCLK in FRC mode.

A.1.40. PICD[1:0] (I/O)

The PICD[1:0] signals are the Execution Control group APIC Data signals. They are used for bidirectional serial message passing on the APIC bus.

A.1.41. PWR_GD (I)

PWR_GD is driven to the Pentium Pro processor by the system to indicate that the clocks and power supplies are within their specification.

This signal is used within the Pentium Pro processor to protect circuits against voltage sequencing issues. While the MTBF of a Pentium Pro processor is on the same order as previous processors without the use of the PWR_GD pin, the use of this signal further increases the Mean Time Between Failures (MTBF) of the Pentium Pro processor component.

This signal will not affect FRC operation.

A.1.42. REQ[4:0]# (I/O)

The REQ[4:0]# signals are the Request Command signals. They are asserted by the current bus owner in both clocks of the Request Phase. In the first clock, the REQa[4:0]# signals define the transaction type to a level of detail that is sufficient to begin a snoop request. In the second clock, REQb[4:0]# signals carry additional information to define the complete transaction type. REQb[4:2]# is reserved. REQb[1:0]# signals transmit LEN[1:0]# (the data transfer length information). In both clocks, REQ[4:0]# and ADS# are protected by parity RP#.

All receiving agents observe the REQ[4:0]# signals to determine the transaction type and participate in the transaction as necessary, as shown in Table A-9.

Table A-9. Transaction Types Defined by REQa#/REQb# Signals

Transaction	REQa[4:0]#					REQb[4:0]#				
	4	3	2	1	0	4	3	2	1	0
Deferred Reply	0	0	0	0	0	x	x	x	x	x
Rsvd *(Ignore)*	0	0	0	0	1	x	x	x	x	x
Interrupt Acknowledge	0	1	0	0	0	DSZ#		x	0	0
Special Transactions	0	1	0	0	0	DSZ#		x	0	1
Rsvd *(Central agent response)*	0	1	0	0	0	DSZ#		x	1	x
Branch Trace Message	0	1	0	0	1	DSZ#		x	0	0
Rsvd *(Central agent response)*	0	1	0	0	1	DSZ#		x	0	1
Rsvd *(Central agent response)*	0	1	0	0	1	DSZ#		x	1	x
I/O Read	1	0	0	0	0	DSZ#		x	LEN#	

Table A-9. Transaction Types Defined by REQa#/REQb# Signals

Transaction	REQa[4:0]#					REQb[4:0]#				
	4	3	2	1	0	4	3	2	1	0
I/O Write	1	0	0	0	1	DSZ#		x	LEN#	
Rsvd *(Ignore)*	1	1	0	0	x	DSZ#		x	x	x
Memory Read & Invalidate	ASZ#		0	1	0	DSZ#		x	LEN#	
Rsvd *(Memory Write)*	ASZ#		0	1	1	DSZ#		x	LEN#	
Memory Code Read	ASZ#		1	D/C#=0	0	DSZ#		x	LEN#	
Memory Data Read	ASZ#		1	D/C#=1	0	DSZ#		x	LEN#	
Memory Write (may not be retried)	ASZ#		1	W/WB#=0	1	DSZ#		x	LEN#	
Memory Write (may be retried)	ASZ#		1	W/WB#=1	1	DSZ#		x	LEN#	

A.1.43. RESET# (I)

The RESET# signal is the Execution Control group reset signal. Asserting RESET# resets all Pentium Pro processors to known states and invalidates their L1 and L2 caches without writing back Modified (M state) lines. RESET# must remain active for one microsecond for a "warm" reset. For a power-on type reset, RESET# must stay active for at least one millisecond after V_{CC} and CLK have reached their proper DC and AC specifications. On observing active RESET#, all bus agents must deassert their outputs within two clocks.

A number of bus signals are sampled at the active-to-inactive transition of RESET# for the power-on configuration. The configuration options are described in Chapter 9, *Configuration* and in every signal description in this chapter.

Unless its outputs are tristated during power-on configuration, after active-to-inactive transition of RESET#, the Pentium Pro processor optionally executes its built-in self-test (BIST) and begins program execution at reset-vector 0_000F_FFF0H or 0_FFFF_FFF0H.

A.1.44. RP# (I/O)

The RP# signal is the Request Parity signal. It is driven by the request initiator in both clocks of the Request Phase. RP# provides parity protection on ADS# and REQ[4:0]#. When a Pentium Pro processor bus agent observes an RP# parity error on any one of the two Request Phase clocks, it must assert AERR# in the Error Phase, provided "AERR# drive" is enabled during the power-on configuration.

A correct parity signal is high if an even number of covered signals are low and low if an odd number of covered signals are low. This definition allows parity to be high when all covered signals are high.

A.1.45. RS[2:0]#(I)

The RS[2:0]# signals are the Response Status signals. They are driven by the response agent (the agent responsible for completion of the transaction at the top of the In-order Queue). Assertion of RS[2:0]# to a non-zero value for one clock completes the Response Phase for a transaction. The response encodings are shown in Table A-10. Only certain response combinations are valid, based on the snoop result signaled during the transaction's Snoop Phase.

Table A-10. Transaction Response Encodings

RS[2:0]#	Description	HITM#	DEFER#
000	Idle State.	NA	NA
001	Retry Response. The transaction is cancelled and must be retried by the initiator.	0	1
010	Defer Response. The transaction is suspended. The defer agent will complete it with a defer reply	0	1
011	Reserved.	0	1
100	Hard Failure. The transaction received a hard error. Exception handling is required.	X	X
101	Normal without data	0	0
110	Implicit Writeback Response. Snooping agent will transfer the modified cache line on the data bus.	1	X
111	Normal with data.	0	0

The RS[2:0]# assertion for a transaction is initiated when all of the following conditions are met:

- All bus agents have observed the Snoop Phase completion of the transaction.
- The transaction is at the top of the In-order Queue.
- RS[2:0]# are sampled in the Idle state

The response driven depends on the transaction as described below:

- The response agent returns a hard-failure response for any transaction in which the response agent observes a hard error.
- The response agent returns a Normal with data response for a read transaction with HITM# and DEFER# deasserted in the Snoop Phase, when the addressed agent is ready to return data and samples inactive DBSY#.

- The response agent returns a Normal without data response for a write transaction with HITM# and DEFER# deasserted in the Snoop Phase, when the addressed agent samples TRDY# active and DBSY# inactive, and it is ready to complete the transaction.
- The response agent must return an Implicit writeback response in the next clock for a read transaction with HITM# asserted in the Snoop Phase, when the addressed agent samples TRDY# active and DBSY# inactive.
- The addressed agent must return an Implicit writeback response in the clock after the following sequence is sampled for a write transaction with HITM# asserted:
  - TRDY# active and DBSY# inactive
  - followed by TRDY# inactive
  - followed by TRDY# active and DBSY# inactive
- The defer agent can return a Deferred, Retry, or Split response anytime for a read transaction with HITM# deasserted and DEFER# asserted.
- The defer agent can return a Deferred or Retry response when it samples TRDY# active and DBSY# inactive for a write transaction with HITM# deasserted and DEFER# asserted.

A.1.46. RSP# (I)

The RSP# signal is the Response Parity signal. It is driven by the response agent during assertion of RS[2:0]#. RSP# provides parity protection for RS[2:0]#.

A correct parity signal is high if an even number of covered signals are low and low if an odd number of covered signals are low. During Idle state of RS[2:0]# (RS[2:0]#=000), RSP# is also high since it is not driven by any agent guaranteeing correct parity.

Pentium Pro processor bus agents can check RSP# at all times and if a parity error is observed, treat it as a protocol violation error. If the BINIT# driver is enabled during configuration, the agent observing RSP# parity error can assert BINIT#.

A.1.47. SMI# (I)

System Management Interrupt is asserted asynchronously by system logic. On accepting a System Management Interrupt, the Pentium Pro processor saves the current state and enters SMM mode. It issues an SMI Acknowledge Bus transaction and then begins program execution from the SMM handler.

A.1.48. SMMEM# (I/O)

The SMMEM# signal is the System Management Mode Memory signal. It is driven on the second clock of the Request Phase on the EXF4#/Ab7# signal. It is asserted by the Pentium Pro processor to indicate that the processor is in System Management Mode and is executing out of SMRAM space.

A.1.49. SPLCK# (I/O)

The SPLCK# signal is the Split Lock signal. It is driven in the second clock of the Request Phase on the EXF3#/Ab6# signal of the first transaction of a locked operation. It is driven to indicate that the locked operation will consist of four locked transactions. Note that SPLCK# is asserted only for locked operations and only in the first transaction of the locked operation.

A.1.50. STPCLK# (I)

The STPCLK# signal is the Stop Clock signal. When asserted, the Pentium Pro processor enters a low power state, the Stop Grant state. The processor issues a Stop Grant Acknowledge special transaction, and stops providing internal clock signals to all units except the bus unit and the APIC unit. The processor continues to snoop bus transactions and service interrupts while in Stop Grant state. When STPCLK# is deasserted, the processor restarts its internal clock to all units and resumes execution. The assertion of STPCLK# has no effect on the bus clock.

STPCLK# is an asynchronous input. In FRC mode, STPCLK# must be synchronous to BCLK.

A.1.51. TCK (I)

The TCK signal is the System Support group Test Clock signal. TCK provides the clock input for the test bus (also known as the test access port). Make certain that TCK is active before initializing the TAP.

A.1.52. TDI(I)

The TDI signal is the System Support group test-data-in signal. TDI transfers serial test data into the Pentium Pro processor. TDI provides the serial input needed for JTAG support.

A.1.53. TDO (O)

The TDO signal is the System Support group test-data-out signal. TDO transfers serial test data out from the Pentium Pro processor. TDO provides the serial output needed for JTAG support.

A.1.54. TMS (I)

The TMS signal is an additional System Support group JTAG-support signal.

A.1.55. TRDY# (I)

The TRDY# signal is the target Ready signal. It is asserted by the target in the Response Phase to indicate that the target is ready to receive write or implicit writeback data transfer. This enables the request initiator or the snooping agent to begin the appropriate data transfer. There will be no data transfer after a TRDY# assertion if a write has zero length indicated in the Request Phase. The data transfer is optional if an implicit writeback occurs for a transaction which writes a full cache line (the Pentium Pro processor will perform the implicit writeback).

TRDY# for a write transaction is driven by the addressed agent when:

- when the transaction has a write or writeback data transfer
- it has a free buffer available to receive the write data
- a minimum of 3 clocks after ADS# for the transaction
- the transaction reaches the top-of-the-In-order Queue
- a minimum of 1 clock after RS[2:0]# active assertion for transaction "*n*-1". (After the transaction reaches the top of the In-order Queue).

TRDY# for an implicit writeback is driven by the addressed agent when:

- transaction has an implicit writeback data transfer indicated in the Snoop Result Phase.
- it has a free cache line buffer to receive the cache line writeback
- if the transaction also has a request initiated transfer, that the request initiated TRDY# was asserted and then deasserted (TRDY# must be deasserted for at least one clock between the TRDY# for the write and the TRDY# for the implicit writeback),
- a minimum of 1 clock after RS[2:0]# active assertion for transaction "*n*-1". (After the transaction reaches the top of the In-order Queue).

TRDY# for a write or an implicit writeback may be deasserted when:

- inactive DBSY# and active TRDY# are observed.
- DBSY# is observed inactive on the clock TRDY# is asserted.
- a minimum of three clocks can be guaranteed between two active-to-inactive transitions of TRDY#
- the response is driven on RS[2:0]#.
- inactive DBSY# and active TRDY# are observed for a write, and TRDY# is required for an implicit writeback.

A.1.56. TRST# (I)

The TRST# signal is an additional System Support group JTAG-support signal.

A.2. SIGNAL SUMMARIES

The following tables list attributes of the Pentium Pro processor output, input, and I/O signals.

Table A-11. Output Signals[1]

Name	Active Level	Clock	Signal Group
FERR#	Low	Asynch	PC compatibility
IERR#	Low	Asynch	Implementation
PRDY#	Low	BCLK	Implementation
TDO	High	TCK	JTAG
THERMTRIP#	Low	Asynch	Implementation

NOTE:

1. Outputs are not checked in FRC mode.

Table A-12. Input Signals[1]

Name	Active Level	Clock	Signal Group	Qualified
A20M#	Low	Asynch	PC compatibility	Always[2]
BPRI#	Low	BLCK	Pentium® Pro processor bus	Always
BR1#	Low	BLCK	Pentium Pro processor bus	Always
BR2#	Low	BLCK	Pentium Pro processor bus	Always
BR3#	Low	BLCK	Pentium Pro processor bus	Always
BCLK	High	-	Pentium Pro processor bus	Always
DEFER#	Low	BLCK	Pentium Pro processor bus	Snoop Phase
FLUSH#	Low	Asynch	PC compatibility	Always[2]
IGNNE#	Low	Asynch	PC compatibility	Always[2]
INIT#	Low	Asynch	Pentium Pro processor bus	Always[2]
INTR	High	Asynch	PC compatibility	APIC disabled mode
LINT[1:0]	High	Asynch	APIC	APIC enabled mode
NMI	High	Asynch	PC compatibility	APIC disabled mode
PICCLK	High	-	APIC	Always
PWR_GD	High	Asynch	Implementation	
PREQ#	Low	Asynch	Implementation	

Table A-12. Input Signals[1] (Contd.)

Name	Active Level	Clock	Signal Group	Qualified
RESET#	High	BCLK	Pentium Pro processor bus	Always
RS[2:0]#	Low	BCLK	Pentium Pro processor bus	Always
RSP#	Low	BCLK	Pentium Pro processor bus	Always
SMI#	Low	Asynch	PC compatibility	
STPCLK#	Low	Asynch	Implementation	
TCK	High	-	JTAG	
TDI		TCK	JTAG	
TMS		TCK	JTAG	
TRST#	Low	Asynch	JTAG	
TRDY#	Low	TCK	Pentium Pro processor bus	Response Phase

NOTES:

1. All asyncronous input signals must be synchronous in FRC
2. Synchronous assertion with active RS[2:0]# guarantees synchronization.

Table A-13. Input/Output Signals (Single Driver)

Name	Active Level	Clock	Signal Group	Qualified
A[35:3]#	Low	BCLK	Pentium® Pro processor bus	ADS#, ADS#+1
ADS#	Low	BCLK	Pentium Pro processor bus	Always
AP[1:0]#	Low	BCLK	Pentium Pro processor bus	ADS#, ADS#+1
ASZ[1:0]#	Low	BCLK	Pentium Pro processor bus	ADS#
ATTR[7:0]#	Low	BCLK	Pentium Pro processor bus	ADS#+1
BE[7:0]#	Low	BCLK	Pentium Pro processor bus	ADS#+1
BR0#	Low	BCLK	Pentium Pro processor bus	Always
BP[3:2]#	Low	BCLK	Pentium Pro processor bus	Always
BPM[1:0]#	Low	BCLK	Pentium Pro processor bus	Always

Table A-13. Input/Output Signals (Single Driver)(Contd.)

Name	Active Level	Clock	Signal Group	Qualified
D[63:0]#	Low	BCLK	Pentium Pro processor bus	DRDY#
DBSY#	Low	BCLK	Pentium Pro processor bus	Always
DEN#	Low	BCLK	Pentium Pro processor bus	ADS# + 1
DEP[7:0]#	Low	BCLK	Pentium Pro processor bus	DRDY#
DID[7:0]#	Low	BCLK	Pentium Pro processor bus	ADS#+1
DSZ[1:0]#	Low	BCLK	Pentium Pro processor bus	ADS#+1
DRDY#	Low	BCLK	Pentium Pro processor bus	Always
EXF[4:0]#	Low	BCLK	Pentium Pro processor bus	ADS#+1
FRCERR	High	BCLK	Implementation	Always
LEN[1:0]#	Low	BCLK	Pentium Pro processor bus	ADS#+1
LOCK#	Low	BCLK	Pentium Pro processor bus	Always
REQ[4:0]#	Low	BCLK	Pentium Pro processor bus	ADS#, ADS#+1
RP#	Low	BCLK	Pentium Pro processor bus	Always
SMMEM#	Low	BCLK	Pentium Pro processor bus	ADS# + 1
SPLCK#	Low	BCLK	Pentium Pro processor bus	ADS# + 1

Table A-14. Input/Output Signals (Multiple Driver)

Name	Active Level	Clock	Signal Group	Qualified
AERR#	Low	BCLK	Pentium® Pro processor bus	ADS# + 3
BNR#	Low	BCLK	Pentium Pro processor bus	Always
BERR#	Low	BCLK	Pentium Pro processor bus	Always
BINIT#	Low	BCLK	Pentium Pro processor bus	Always
HIT#	Low	BCLK	Pentium Pro processor bus	Always
HITM#	Low	BCLK	Pentium Pro processor bus	Always
PICD[1:0]	High	PICCLK	APIC	Always

UNITED STATES, Intel Corporation
2200 Mission College Blvd., P.O. Box 58119, Santa Clara, CA 95052-8119
Tel: +1 408 765-8080

JAPAN, Intel Japan K.K.
5-6 Tokodai, Tsukuba-shi, Ibaraki-ken 300-26
Tel: + 81-29847-8522

FRANCE, Intel Corporation S.A.R.L.
1, Quai de Grenelle, 75015 Paris
Tel: +33 1-45717171

UNITED KINGDOM, Intel Corporation (U.K.) Ltd.
Pipers Way, Swindon, Wiltshire, England SN3 1RJ
Tel: +44 1-793-641440

GERMANY, Intel GmbH
Dornacher Strasse 1
85622 Feldkirchen/ Muenchen
Tel: +49 89/99143-0

HONG KONG, Intel Semiconductor Ltd.
32/F Two Pacific Place, 88 Queensway, Central
Tel: +852 2844-4555

CANADA, Intel Semiconductor of Canada, Ltd.
190 Attwell Drive, Suite 500
Rexdale, Ontario M9W 6H8
Tel: +416 675-2438

Printed in USA/7K/0396/NCG/PMG